

IN THE CLAIMS

For the convenience of the Examiner, all claims of the present Application are shown whether or not an amendment has been made.

1. (Previously Amended) A method for traversing a boundary in a distributed processing environment, comprising:

storing connection protocol information in a connection properties table at a client network for each boundary which may be traversed by the client network;

receiving a request from a client object on the client network for access to a server object on a server network, the server network having a server network boundary;

locating an entry in the connections property table corresponding to the requested server object;

formatting a boundary traversal key from the connection protocol information associated with the located entry in the connection properties table, the boundary traversal key including information to traverse a boundary controlling access to the server network; and

forwarding the request for access and the boundary traversal key to the boundary controlling access to the server network.

2. (Original) The method of Claim 1, further comprising determining a connection type from the located entry in the connections property table.

3. (Original) The method of Claim 1, further comprising:

passing the request for access to an object request broker after the client network determines that the request for access is to an object residing outside the client network.

4. (Original) The method of Claim 3, wherein the object request broker locates the entry, formats the boundary traversal key, and forwards the request for access and the boundary traversal key to the server network.

5. (Original) The method of Claim 1, wherein storing connection protocol information includes storing a boundary identifier, a connection type, authentication information, and connection attributes in the connection properties table.

6. (Original) The method of Claim 5, wherein locating an entry includes matching an internet protocol address for the server object to the boundary identifiers stored in the connection properties table.

7. (Original) The method of Claim 5, wherein locating an entry includes matching a domain name for the server object to the boundary identifiers stored in the connection properties table.

8. (Original) The method of Claim 5, wherein locating an entry includes matching a port address for the server object to the boundary identifiers stored in the connection properties table.

9. (Original) The method of Claim 5, wherein formatting the boundary traversal key includes building the boundary traversal key from the authentication information and the connection attributes in a format defined by the connection type.

10. (Original) The method of Claim 1, wherein forwarding the request includes forwarding the request for access and the boundary traversal key to the server network boundary.

11. (Original) The method of Claim 1, further comprising:

receiving the request for access and the boundary traversal key at the server network boundary;

allowing access to the server object if the server network boundary accepts the boundary traversal key; and

denying access to the server object if the server network boundary rejects the boundary traversal key.

12. (Previously Amended) A distributed computing system, comprising:

a client object on a first network operable to request access to a server object on a second network;

a third network connecting the first network to the second network;

a boundary device controlling access to the second network;

a connections properties table in the first network and including an entry for each of one or more second networks accessible by the first network, the connections properties table including connection protocol information for accessing the one or more second networks;

a connection manager operable to generate a boundary traversal key for requests for access to server objects that have a corresponding entry in the connections properties table, the boundary traversal key generated from the corresponding connection protocol information, the boundary traversal key including information to traverse the boundary device controlling access to the second network.

13. (Original) The system of Claim 12, further comprising a default connection manager operable to establish a connection between the client object and the server object using a default protocol for requests for access to server objects that do not have a corresponding entry in the connection properties table.

14. (Original) The system of Claim 12, wherein the third network is an Internet.

15. (Original) The system of Claim 12, further comprising an object request broker operable to facilitate communications between the client object and the server object across the third network.

16. (Original) The system of Claim 15, wherein the connection manager is part of the object request broker.

17. (Original) The system of Claim 12, wherein the connection properties table includes:

a boundary identifier for identifying the server object on the second network;

a connection type for identifying the type of connection protocol used by the second network;

authentication information for providing identity and credential information to the second network; and

attributes for providing boundary traversal key information to the second network.

18. (Original) The system of Claim 12, wherein the connection properties table is stored in a private directory on the first network.

19. (Original) The system of Claim 17, wherein the boundary traversal key is generated from the authentication information and the attributes from an entry in the connection properties table corresponding to the server object on the second network.

20. (Original) The system of Claim 17, wherein the boundary identifier is an identifier selected from the group consisting of an internet protocol address, an internet protocol address range, a partial internet protocol address, a domain name, a partial domain name, a port address and a port address range.

21. (Original) The system of Claim 17, wherein the connection type indicates a TCP/IP connection, an SSL connection, an HTTP Tunneling connection, or a UDP/IP connection.

22. (Original) The system of Claim 17, wherein the authentication information includes a user identification and a password.

23. (Previously Amended) A distributed processing system with transparent boundary traversal, comprising:

- a client system operable to request access to a plurality of server systems, at least one of the server systems having a boundary device for controlling access to the server system by the client system;

- a connection properties table stored in a private directory on the client system, the connection properties table including:

 - an identification range for identifying the at least one server system having the boundary device;

 - a boundary type for identifying a type of the boundary device;

 - authentication information for uniquely identifying the client system to the boundary device and a requested server system; and

 - attributes for providing traversal information required by the boundary device;

- a boundary traversal key generator operable to generate a boundary traversal key for gaining access to the requested server system through the boundary device, the boundary traversal key generated from the connection properties table in response to the boundary traversal key generator locating an entry matching the requested server system.

24. (Original) The system of Claim 23, further comprising a network for connecting the client system to the server system.

25. (Original) The system of Claim 24, further comprising an object request broker operable to facilitate communications between the client object and the server object across the network.

26. (Original) The system of Claim 24, wherein the network is an internet.

27. (Original) The system of Claim 25, wherein the boundary traversal key generator is part of the object request broker.